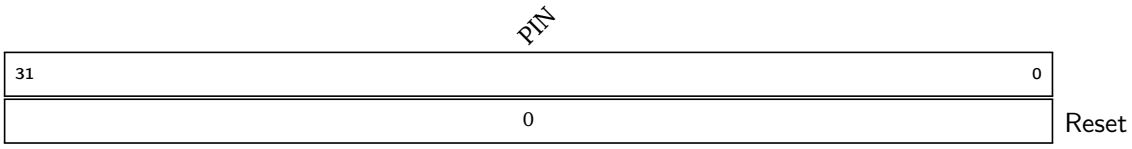
C I/O registers

This is a list of currently supported I/O addresses. The default start address is 0xF0000000. The offset is given in bytes. Note that the I/O can only be accesses on 4-byte boundaries and on word size accesses.

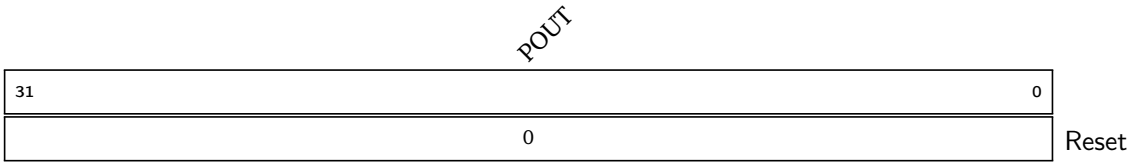
C.1 GPIOA – General Purpose I/O



Register C.1: PORT A INPUT REGISTER GPIOA_PIN *(0x000)*

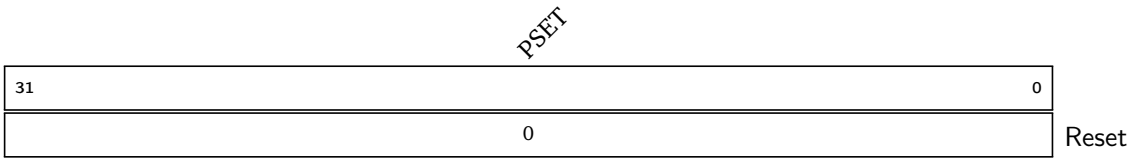
PIN Port pin input values.

Note: This I/O register can only be read. Writes are ignored.



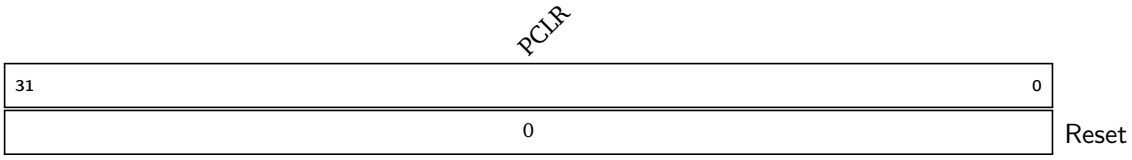
Register C.2: PORT A OUTPUT REGISTER GPIOA_POUT *(0x004)*

POUT Port pin output values. Write: The data is written to the output pins. Read: The last entered data is read back.



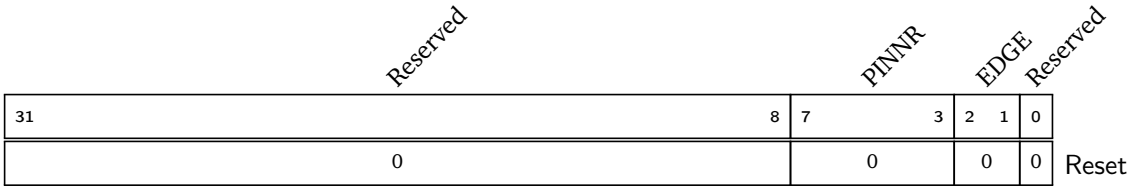
Register C.3: PORT A OUTPUT SET REGISTER GPIOA_PSET *(0x008)*

PSET Set atomically bits in the GPIOA output register. If a bit is 1, the corresponding bit in GPIOA_POUT register is set. If a bit is 0, there is no change to the correnspondingGPIOA_POUT bit. This register is write-only. Reads return all-zero bits.



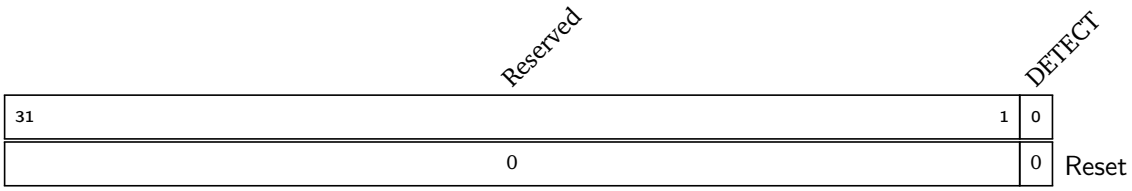
Register C.4: PORT A OUTPUT CLEAR REGISTER GPIOA_PCLR *(0x00c)*

PCLR Clear atomically bits in the GPIOA output register. If a bit is 1, the corresponding bit in GPIOA_POUT register is cleared. If a bit is 0, there is no change to the corrensponding GPIOA_POUT bit. This register is write-only. Reads return all-zero bits.



Register C.5: EXTERNAL INPUT INTERRUPT CONTROL REGISTER GPIOA_EXTC *(0x018)*

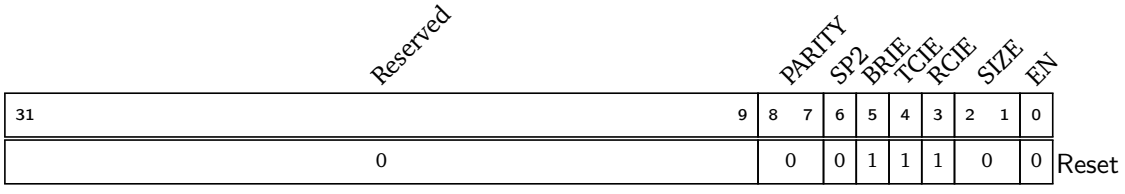
PINNR Port pin number as input source.
EDGE Edge selection: 00 = off, 01 = rising, 10 = falling, 11 = both.



Register C.6: EXTERNAL INPUT INTERRUPT STATUS REGISTER GPIOA_EXTS *(0x01c)*

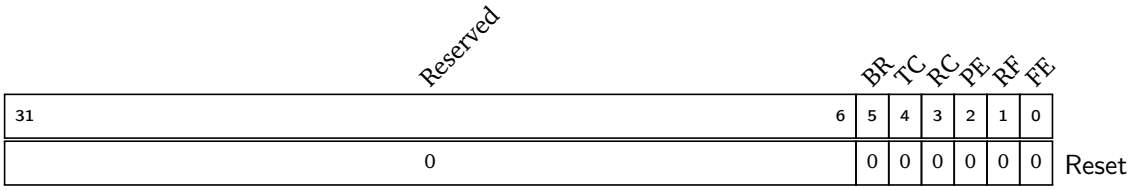
DETECT Edge detected. Must be cleared to reset the pending interrupt.

C.2 UART1 – Universal Asynchronous Receiver/Transmitter



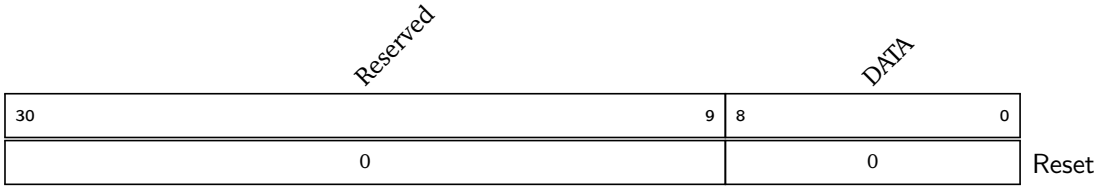
Register C.7: UART1 CONTROL REGISTER UART1_CTRL *(0x100)*

PARITY	00: none, 10: even, 11: odd.
SP2	0: one stop bit, 1: two stop bits.
BRIE	BREAK received interrupt enable.
TCIE	Transmit character interrupt enable.
RCIE	Receive character interrupt enable.
SIZE	00: 8 bits, 10: 9 bits, 11: 7 bits, excluding the parity.
EN	Enable UART1 (both receiver and transmitter)



Register C.8: UART1 STATUS REGISTER UART1_STAT *(0x104)*

BR	BREAK condition detected. A BREAK is a stream of null bits for the duration of 1 start bit + number of data bits + 1 stop bit.
TC	Transmit completed. Set directly to 1 when a character was transmitted. Automatically cleared when writing new character to the data register or when writing 0 in the TC bit in UART1_STAT.
RC	Receive completed. Set to 1 when a character was received. Automatically cleared when data register is read or when writing 0 in the RC bit in UART1_STAT.
PE	Parity error. Set to 1 if parity is enabled and there is a parity error while receiving. Automatically cleared when data register is read or when writing 0 in the PE bit in UART1_STAT.
RF	Receive failed. Set to 1 when failed receiving (invalid start bit). Automatically cleared when data register is read or when writing 0 in the RF bit in UART1_STAT.
FE	Frame error. Set to 1 when a low is detected at the position of the (first) stop bit. Automatically cleared when data register is read or writing a 0 in the FE bit in UART1_STAT.

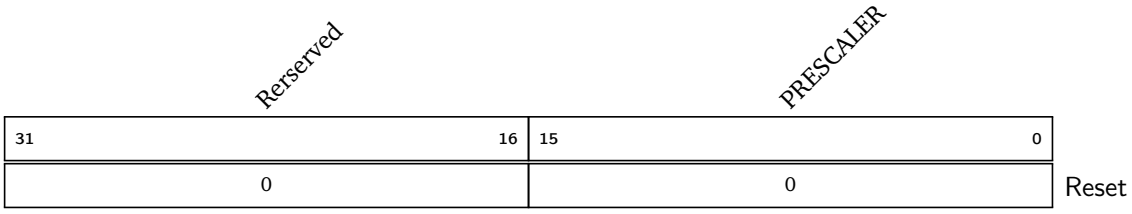


Register C.9: UART1 DATA REGISTER UART1_DATA *(0x108)*

DATA	The data to be transmitted or received.

Write: The data is written to an internal buffer and transmitted.

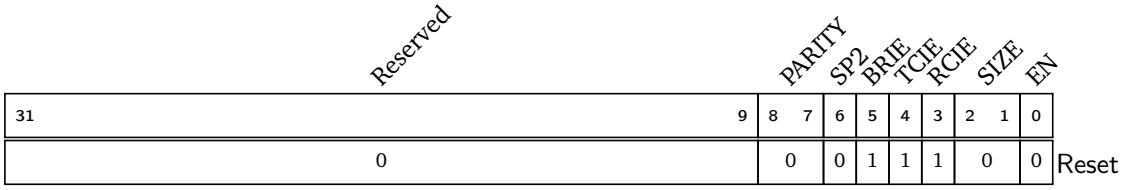
Read: The last received data is read.
Size depends on the SIZE field in the UART1 Control Register.



Register C.10: UART1 BAUD RATE REGISTER UART1_BAUD *(0x10c)*

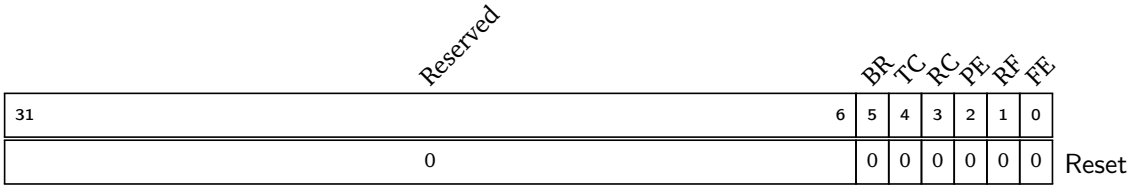
PRESCALER $\text{Baud rate} = \dfrac{f_{system}}{\text{prescaler} + 1}$

C.3 UART2 – Universal Asynchronous Receiver/Transmitter



Register C.11: UART2 CONTROL REGISTER UART2_CTRL *(0xb00)*

PARITY	00: none, 10: even, 11: odd.
SP2	0: one stop bit, 1: two stop bits.
BRIE	BREAK received interrupt enable.
TCIE	Transmit character interrupt enable.
RCIE	Receive character interrupt enable.
SIZE	00: 8 bits, 10: 9 bits, 11: 7 bits, excluding the parity.
EN	Enable UART2 (both receiver and transmitter)



Register C.12: UART2 STATUS REGISTER UART2_STAT *(0xb04)*

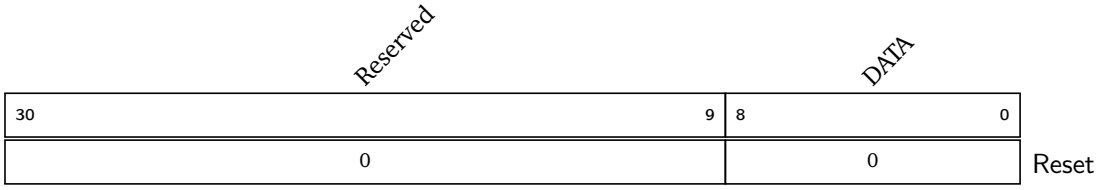
BR	BREAK condition detected. A BREAK is a stream of null bits for the duration of 1 start bit + number of data bits + 1 stop bit.
TC	Transmit completed. Set directly to 1 when a character was transmitted. Automatically cleared when writing new character to the data register or when writing 0 in the TC bit in UART2_STAT.

RC Receive completed. Set to 1 when a character was received. Automatically cleared when data register is read or when writing 0 in the RC bit in UART2_STAT.

PE Parity error. Set to 1 if parity is enabled and there is a parity error while receiving. Automatically cleared when data register is read or when writing 0 in the PE bit in UART2_STAT.

RF Receive failed. Set to 1 when failed receiving (invalid start bit). Automatically cleared when data register is read or when writing 0 in the RF bit in UART2_STAT.

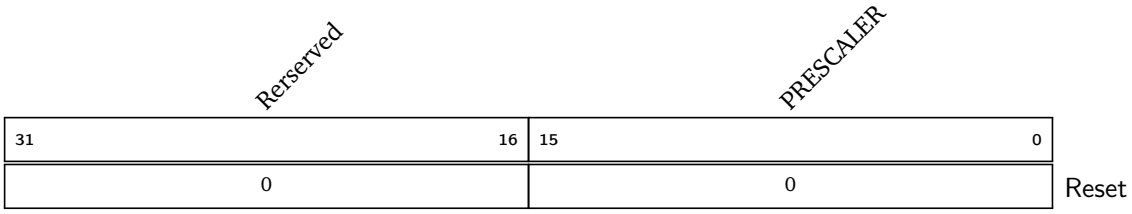
FE Frame error. Set to 1 when a low is detected at the position of the (first) stop bit. Automatically cleared when data register is read or writing a 0 in the FE bit in UART2_STAT.



Register C.13: UART2 DATA REGISTER UART2_DATA *(0xb08)*

DATA The data to be transmitted or received.

Write: The data is written to an internal buffer and transmitted.
Read: The last received data is read.
Size depends on the SIZE field in the UART2 Control Register.

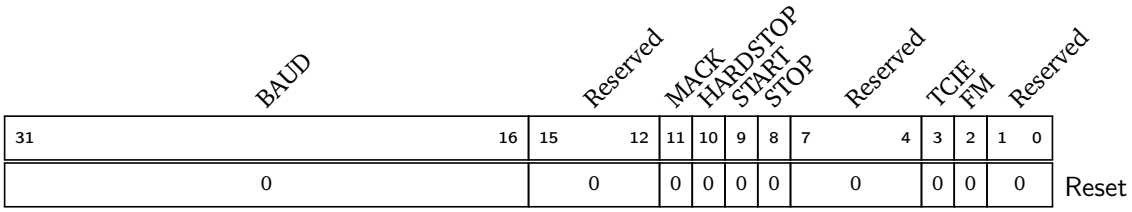


Register C.14: UART2 BAUD RATE REGISTER UART2_BAUD *(0x10c)*

PRESCALER $\text{Baud rate} = \dfrac{f_{system}}{\text{prescaler} + 1}$

C.4 I2C1 – Inter-Integrated Circuit master-only controller

General purpose I^2C peripheral, with programmable baud rate prescaler, start- and stop-bit generation, support for clock stretching, no arbitration, Standard mode (Sm) and Fast mode (Fm) only.



Register C.15: I2C1 CONTROL REGISTER I2C1_CTRL *(0x200)*

BAUD Baud rate prescaler. Number of system clock pulses minus 1 for **one-half** bit time (Sm) or **one-third** bit time (Fm). Note: because of the 50 MHz system frequency, the lowest I^2C clock frequency is 763 Hz (Sm).

MACK Set to 1 to acknowledge a reception by the master. Must only be used when receiving.

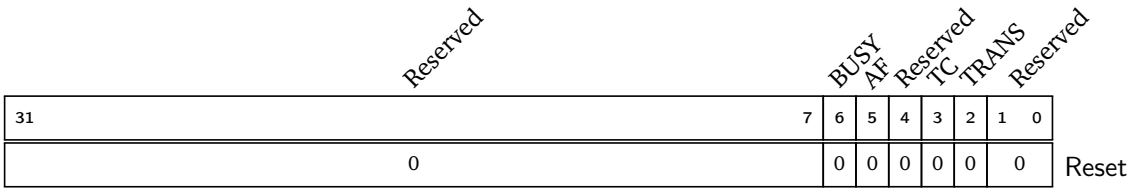
HARDSTOP Set to 1 to just generate a STOP condition. Useful after addressing a target that didn't respond. Cleared by hardware.

START Send a START before next byte send. Cleared by hardware when transmission ends.

STOP Send a STOP after next byte send or received. Cleared by hardware when transmission ends.

TCIE Transmission Complete interrupt enable.

FM 0: Standard mode 1:1 (SCL 1/2 low, 1/2 high)
1: Fast mode 2:1 (SCL 2/3 low, 1/3 high)



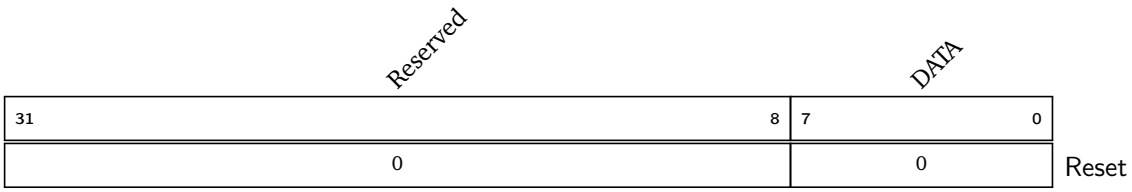
Register C.16: I2C1 STATUS REGISTER I2C1_STAT *(0x204)*

BUSY Set to 1 when SDA or SCL is low, set to 0 when STOP condition is detected, independent of the I2C1 device.

AF Acknowledge Fail, set when no target responded. Cleared by hardware when `I2C1_DATA` is accessed.

TC Transmission Complete, including START or STOP, if any. Cleared by hardware when `I2C1_DATA` is accessed.

TRANS Indicates transmitting (1) or not (0) by this controller.



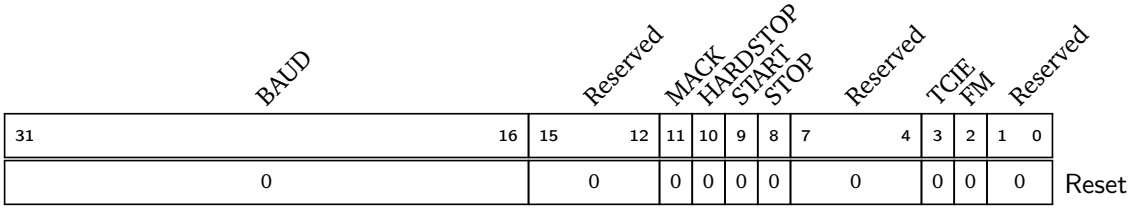
Register C.17: I2C1 DATA REGISTER I2C1_DATA *(0x208)*

DATA	Data to be transmitted or received.

Write: The data is written to an internal buffer and transmitted.
Read: The last received data is read.

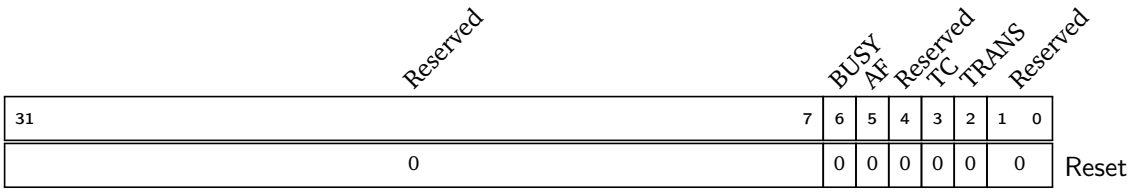
C.5 I2C2 – Inter-Integrated Circuit master-only controller

General purpose I^2C peripheral, with programmable baud rate prescaler, start- and stop-bit generation, support for clock stretching, no arbitration, Standard mode (Sm) and Fast mode (Fm) only.



Register C.18: I2C2 CONTROL REGISTER I2C2_CTRL *(0x300)*

BAUD	Baud rate prescaler. Number of system clock pulses minus 1 for **one-half** bit time (Sm) or **one-third** bit time (Fm). Note: because of the 50 MHz system frequency, the lowest I^2C clock frequency is 763 Hz (Sm).
MACK	Set to 1 to acknowledge a reception by the master. Must only be used when receiving.
HARDSTOP	Set to 1 to just generate a STOP condition. Useful after addressing a target that didn't respond. Cleared by hardware.
START	Send a START before next byte send. Cleared by hardware when transmission ends.
STOP	Send a STOP after next byte send or received. Cleared by hardware when transmission ends.
TCIE	Transmission Complete interrupt enable.
FM	0: Standard mode 1:1 (SCL 1/2 low, 1/2 high) 1: Fast mode 2:1 (SCL 2/3 low, 1/3 high)



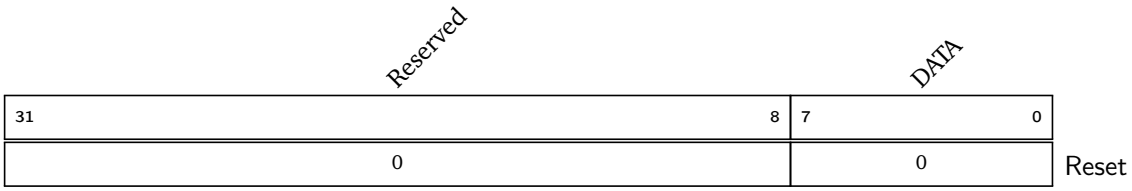
Register C.19: I2C2 STATUS REGISTER I2C2_STAT *(0x304)*

BUSY Set to 1 when SDA or SCL is low, set to 0 when STOP condition is detected, independent of the I2C2 device.

AF Acknowledge Fail, set when no target responded. Cleared by hardware when `I2C2_DATA` is accessed.

TC Transmission Complete, including START or STOP, if any. Cleared by hardware when `I2C2_DATA` is accessed.

TRANS Indicates transmitting (1) or not (0) by this controller.



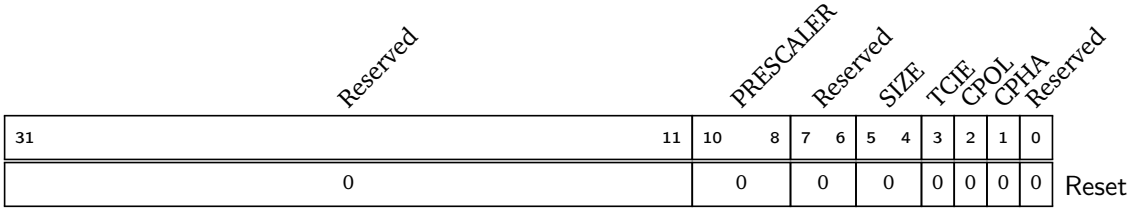
Register C.20: I2C2 DATA REGISTER I2C2_DATA *(0x308)*

DATA Data to be transmitted or received.

Write: The data is written to an internal buffer and transmitted.
Read: The last received data is read.

C.6 SPI1 – Serial Peripheral Interface

General purpose SPI master peripheral, with prescaler, 8/16/24/32 bits data exchange and interrupt.



Register C.21: SPI1 CONTROL REGISTER SPI1_CTRL *(0x400)*

PRESCALER **000** /2
001 /4

010 /8
011 /16
100 /32
101 /64
110 /128
111 /256

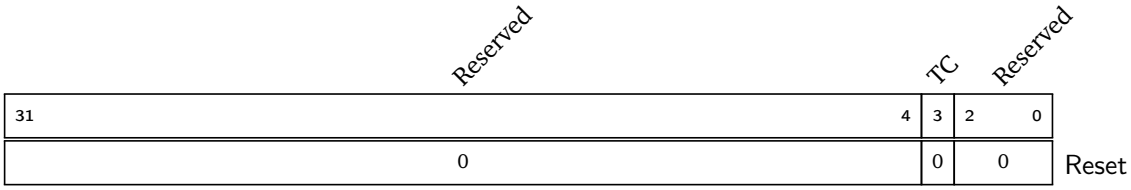
Note: because of the 50 MHz system frequency, the lowest SPI clock frequency is 195.3125 kHz.

SIZE **00** 8 bits
01 16 bits
10 24 bits
11 32 bits

TCIE Transfer complete interrupt enable

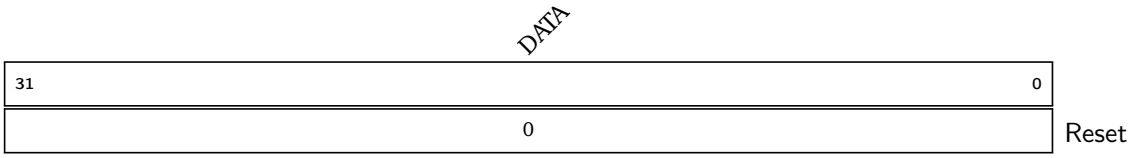
CPOL Clock polarity

CPHA Transfer phase



Register C.22: SPI1 STATUS REGISTER SPI1_STAT *(0x404)*

TC Transfer complete



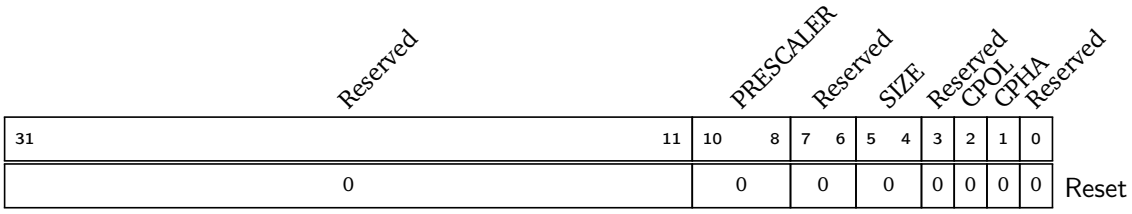
Register C.23: SPI1 DATA REGISTER SPI1_DATA *(0x408)*

DATA Data to be transmitted or received.

Write: The data is written to an internal buffer and transmitted.
Read: The last received data is read.
Data size depends on the SIZE field in the SPI1 Control Register. Data is right aligned.

C.7 SPI2 – Serial Peripheral Interface

General purpose SPI master peripheral, with prescaler, 8/16/24/32 bits data exchange and interrupt.



Register C.24: SPI2 CONTROL REGISTER SPI2_CTRL *(0x500)*

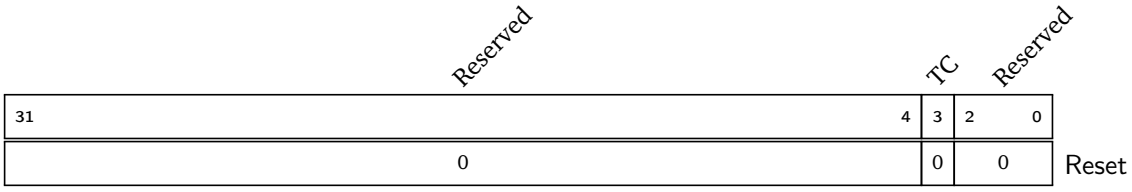
PRESCALER
000 /2
001 /4
010 /8
011 /16
100 /32
101 /64
110 /128
111 /256

Note: because of the 50 MHz system frequency, the lowest SPI clock frequency is 195.3125 kHz.

SIZE
00 8 bits
01 16 bits
10 24 bits
11 32 bits

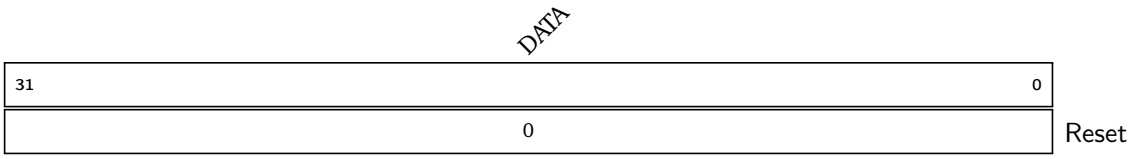
CPOL Clock polarity

CPHA Transfer phase



Register C.25: SPI2 STATUS REGISTER SPI2_STAT *(0x504)*

TC Transfer complete



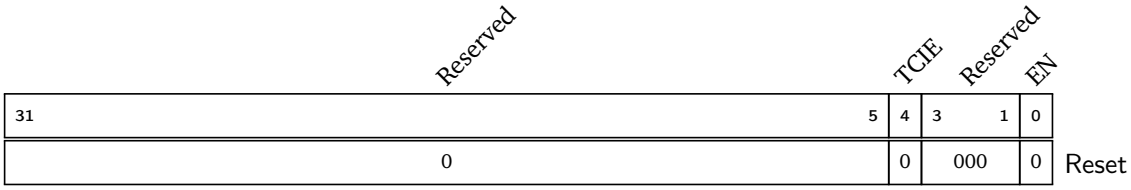
Register C.26: SPI2 DATA REGISTER SPI2_DATA *(0x508)*

DATA Data to be transmitted or received.

Write: The data is written to an internal buffer and transmitted.
Read: The last received data is read.
Data size depends on the SIZE field in the SPI2 Control Register. Data is right aligned.

C.8 TIMER1 – a simple timer

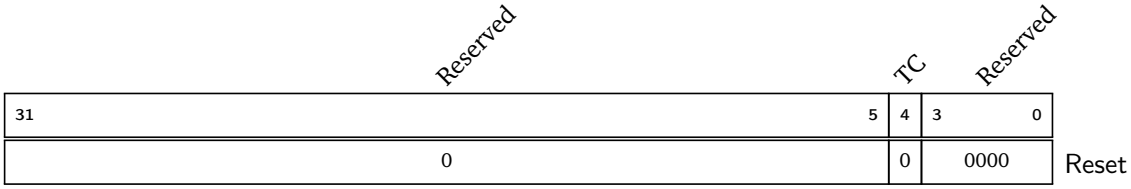
Simple 32-bit timer peripheral for time base generation, with interrupt.



Register C.27: TIMER1 CONTROL REGISTER TIMER1_CTRL *(0x600)*

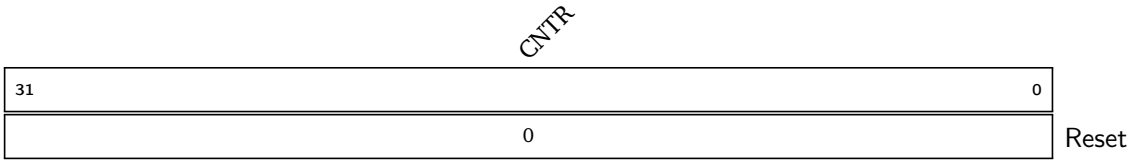
EN Enable the timer

TCIE Timer compare match interrupt enable



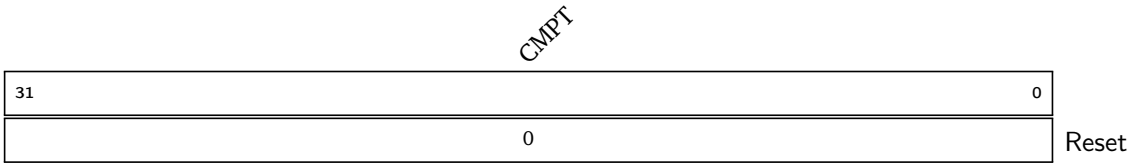
Register C.28: TIMER1 STATUS REGISTER TIMER1_STAT *(0x604)*

TCI Timer compare match. Set to 1 on compare match between the timer Count register and the Compare Match register. Must be cleared by software by writing a 0.



Register C.29: TIMER1 COUNT REGISTER TIMER1_CNTR *(0x608)*

CNTR This register holds the counted clock pulses on the timer. This register may be written by software.

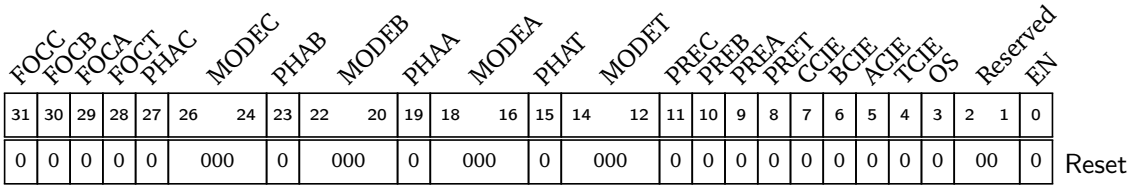


Register C.30: TIMER1 COMPARE TIMER T REGISTER TIMER1_CMPT *(0x60c)*

CMPT This register holds the value at which the counter register is compared. On CNTR compares to greater than or equal to CMPT, the counter register will be cleared and the TCI flag will be set (both in the next clock cycle).

C.9 TIMER2 – a more elaborate timer

General purpose 16-bit timer with Output Compare, PWM generation and Input Capture capabilities, preload and interrupt (one vector).



Register C.31: TIMER2 CONTROL REGISTER TIMER2_CTRL *(0x700)*

FOCC Force Output Compare match C.
FOCB Force Output Compare match B.
FOCA Force Output Compare match A.
FOCT Force Output Compare match T.
PHAC Register C start phase.
MODEC Register C mode.
PHAB Register B start phase.
MODEB Register B mode.
PHAA Register A start phase.
MODEA Register A mode.
PHAT Register T start phase.

MODET Register T mode.
PREC Enable compare register C preload.
PREB Enable compare register B preload.
PREA Enable compare register A preload.
PRET Enable compare register T preload.
CCIE Timer compare match/input capture C interrupt enable.
BCIE Timer compare match/input capture B interrupt enable.
ACIE Timer compare match/input capture A interrupt enable.
TCIE Timer compare match T interrupt enable.
OS One-shot mode.
EN Enable the timer.

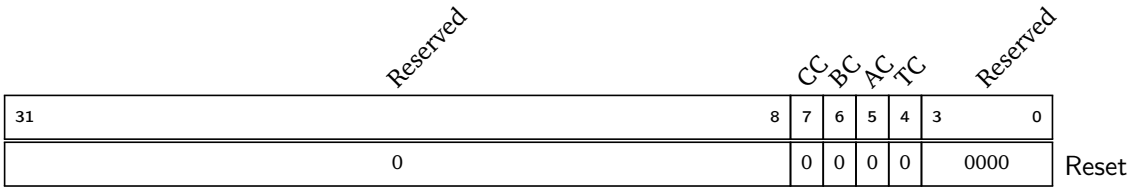
If none of the FOC*x* bits are 1, MODET and MODEA/B/C have the following meaning:

000 Output off.
001 Toggle on compare match.
010 Set high on compare match.
011 Set low on compare match.
100 Edge-aligned PWM (only A/B/C, for T not allowed).
101 Reserved.
110 Input capture positive edge (only A/B/C, for T not allowed).
111 Input capture negative edge (only A/B/C, for T not allowed).

If at least one of the FOC*x* bits is 1, MODET and MODEA/B/C have the following meaning:

000 Not used.
001 Toggle output compare.
010 Set high output compare.
011 Set low output compare.
100 not allowed.
101 not allowed.
110 not allowed.
111 not allowed.

In this case, the CTRL register is not written and keeps its original setting.



Register C.32: TIMER2 STATUS REGISTER TIMER2_STAT *(0x704)*

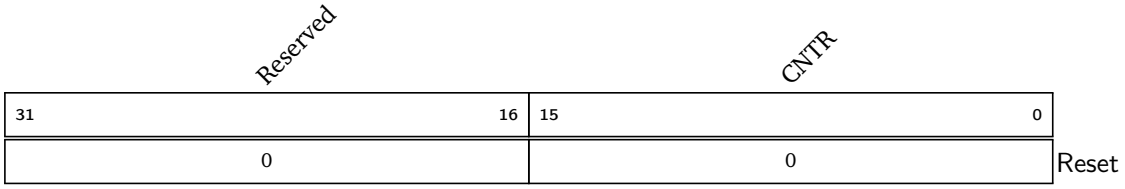
CC Timer compare match C. Set to 1 on compare match between the timer Count register and the Compare Match C register. Set

on input capture on detecting selected edge. Must be cleared by software by writing a 0.

BC Timer compare match B. Set to 1 on compare match between the timer Count register and the Compare Match B register. Set on input capture on detecting selected edge. Must be cleared by software by writing a 0.

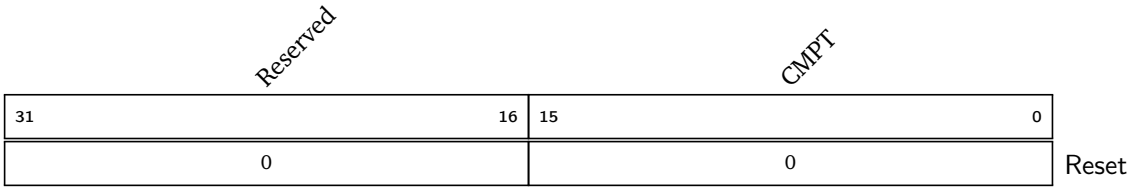
AC Timer compare match A. Set to 1 on compare match between the timer Count register and the Compare Match A register. Set on input capture on detecting selected edge. Must be cleared by software by writing a 0.

TC Timer compare match T. Set to 1 on compare match between the timer Count register and the Compare Match T register. Must be cleared by software by writing a 0.



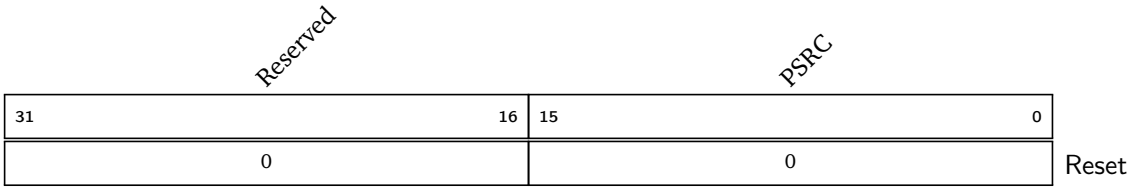
Register C.33: TIMER2 COUNT REGISTER TIMER2_CNTR *(0x708)*

CNTR This register holds the counted clock pulses on the timer. This register may be written by software. Rolls over when CNTR compare greater than or equal to CMPT on the next clock cycle.



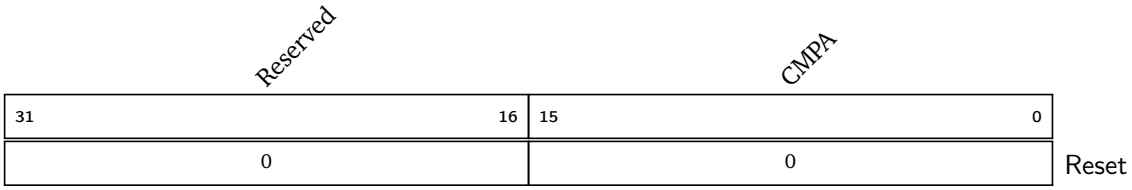
Register C.34: TIMER2 COMPARE TIMER T REGISTER TIMER2_CMPT *(0x70c)*

CMPT This register holds the value at which the Count register is compared. On CNTR compares to greater than or equal to CMPT, the Count register will be cleared and the TC flag will be set (both in the next clock cycle).



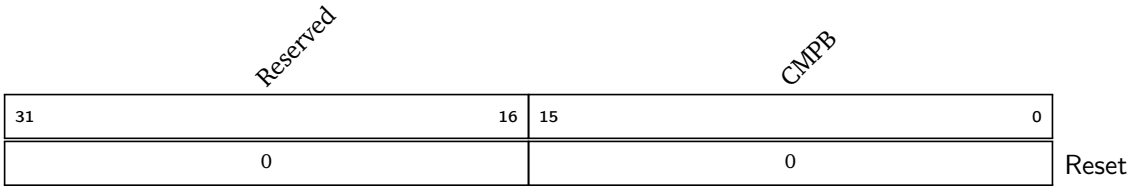
Register C.35: TIMER2 PRESCALER REGISTER TIMER2_PRSC *(0x710)*

PRSC This register holds the prescaler of the timer. This register may be written by software. Whenever the internal prescaler is equal to or greater than this register, the internal prescaler is reset. This register should only be written when the timer is stopped. Writing this register resets the internal prescaler.



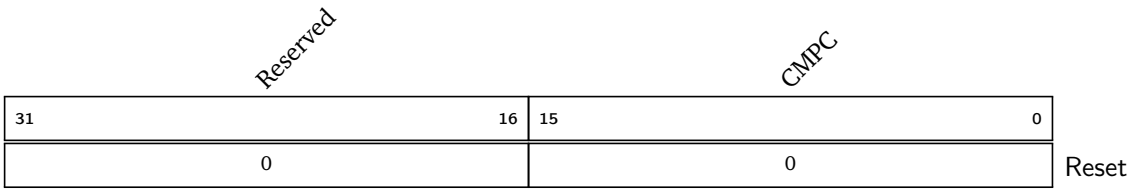
Register C.36: TIMER2 COMPARE TIMER A REGISTER TIMER2_CMPA *(0x714)*

CMPA For Output Compare: This register holds the value at which the Count register is compared. On CNTR compares to greater than or equal to CMPA, the ACI flag will be set in the next clock cycle. For Input Capture: The value of CNTR is copied to CMPA on detecting the selected edge, and the AC flag is set.



Register C.37: TIMER2 COMPARE TIMER B REGISTER TIMER2_CMPB *(0x718)*

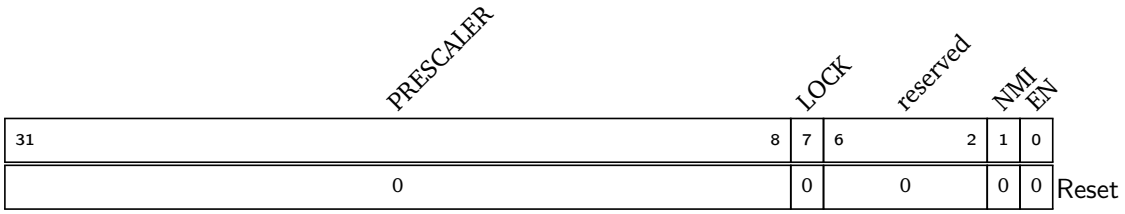
CMPB For Output Compare: This register holds the value at which the Count register is compared. On CNTR compares to greater than or equal to CMPB, the BCI flag will be set in the next clock cycle. For Input Capture: The value of CNTR is copied to CMPB on detecting the selected edge, and the BC flag is set.



Register C.38: TIMER2 COMPARE TIMER C REGISTER TIMER2_CMPC *(0x71c)*

CMPC For Output Compare: This register holds the value at which the Count register is compared. On CNTR compares to greater than or equal to CMPC, the CCI flag will be set in the next clock cycle. For Input Capture: The value of CNTR is copied to CMPC on detecting the selected edge, and the CC flag is set.

C.10 WDT – Watchdog Timer



Register C.39: WATCHDOG CONTROL REGISTER WDT_CTRL *(0x800)*

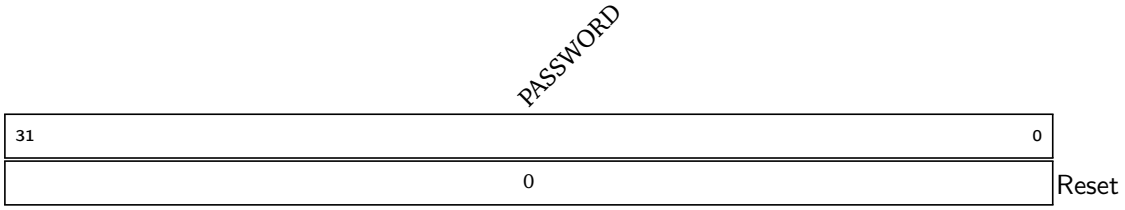
PRESCALER 24-bit prescaler. The timeout time is computed with

$$t = \frac{\text{Prescaler} \cdot 256}{f_{cpu}}$$

LOCK If set to 1, the control register is locked. Watchdog write accesses trigger a system reset or NMI.

NMI If set to 1, the watchdog triggers an NMI on timeout, otherwise the watchdog triggers a system reset on timeout.

EN If set to 1, the WDT starts counting.

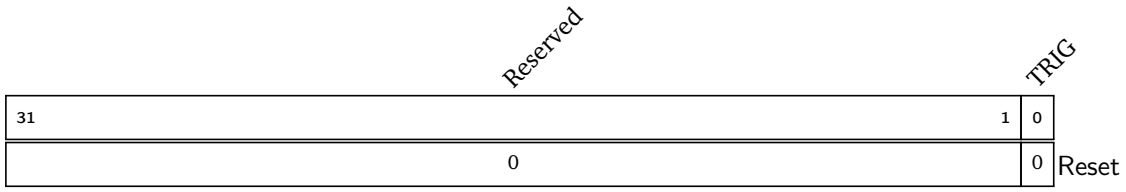


Register C.40: WATCHDOG TRIGGER REGISTER WDT_TRIG *(0x804)*

Note: this register must be written with the WDT password to reset the watchdog. No actual value is written in this register. Reads return all zero bits.

C.11 MSI – Machine Software Interrupt

Note: MSI has to be enabled by writing a 1 to `mie.MSIE`.



Register C.41: MSI TRIGGER REGISTER MSI_TRIG *(0x900)*

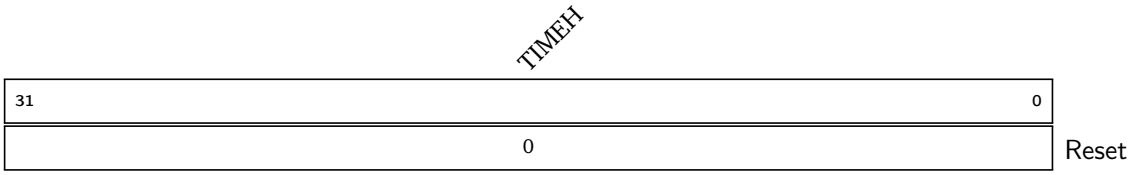
TRIG Writing a 1 to this field will trigger an MSI. Writing a 0 will disarm the trigger.

C.12 MTIME – RISC-V system timer

Note: the external timer interrupt has to be enabled by writing a 1 to `mie.MTIE`. Note: the external timer will assert a pending interrupt if TIMEH:TIME (viewed as a 64-bit register) is greater than or equal to TIMECMPH:TIMECPM (viewed as a 64-bit register). To negate the pending interrupt, set TIMECMPH:TIMECMP to a higher value than TIMEH:TIME. The TIMEH:TIME registers count the number of micro seconds since last reset. As such, the system clock frequency must be a integer multiple of 1 MHz.

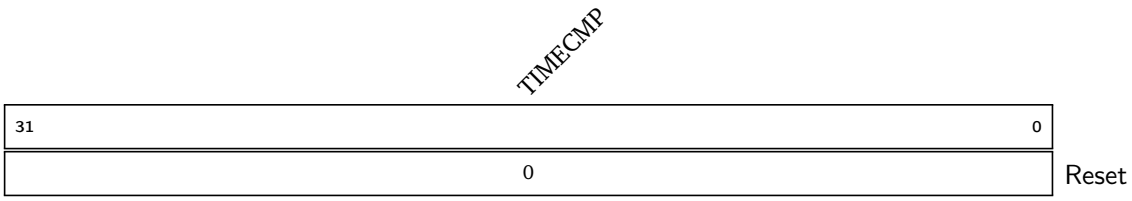
Register C.42: TIME EXTERNAL TIMER REGISTER TIME *(0xa00)*

TIME This register holds the low 32 bits of the external timer. Currently read-only.



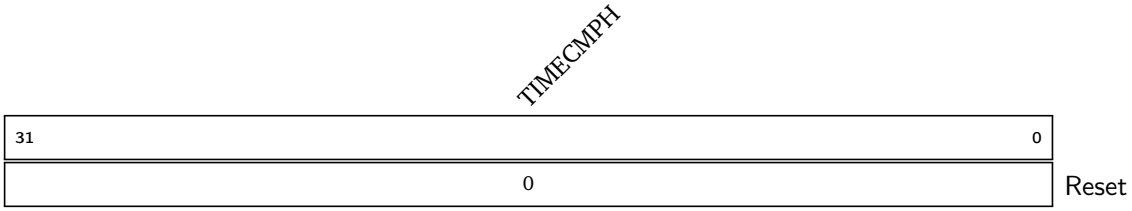
Register C.43: TIMEH EXTERNAL TIMER REGISTER TIME *(0xa04)*

TIMEH This register holds the upper 32 bits of the external timer. Currently read-only.



Register C.44: TIMECMP EXTERNAL TIMER COMPARE REGISTER TIMECMP *(0xa08)*

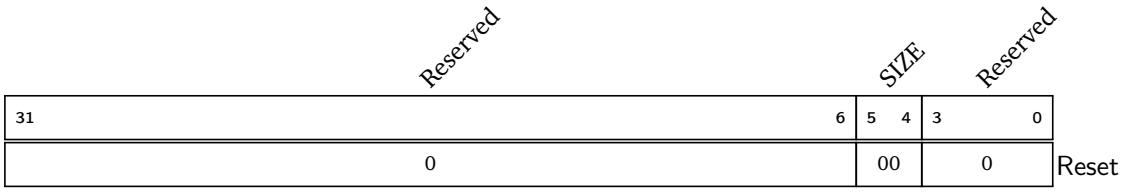
TIMECMP This register holds the low 32 bits of the external timer compare register.



Register C.45: TIMECMPH EXTERNAL TIMER COMPARE REGISTER TIMECMP *(0xa0c)*

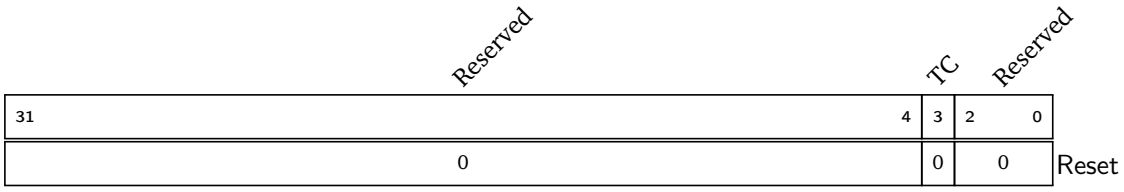
TIMECMPH This register holds the upper 32 bits of the external timer compare register.

C.13 CRC – Cyclic Redundancy Check



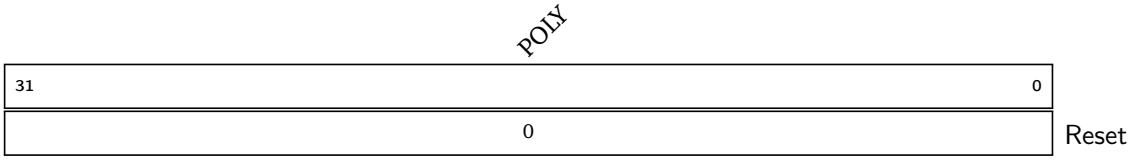
Register C.46: CRC CONTROL REGISTER CRC_CTRL *(0xc00)*

SIZE 00 = 32 bits, 01 = 24 bits, 10 = 16 bits, 11 = 8 bits.



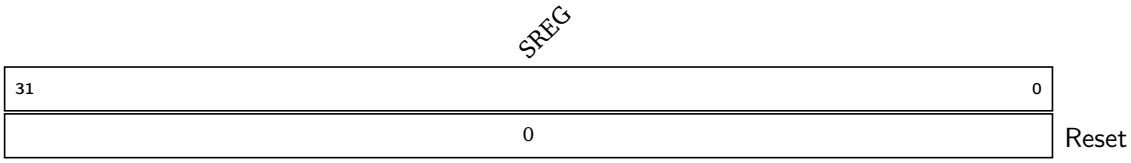
Register C.47: CRC STATUS REGISTER CRC_STAT *(0xc04)*

TC Transaction Complete.



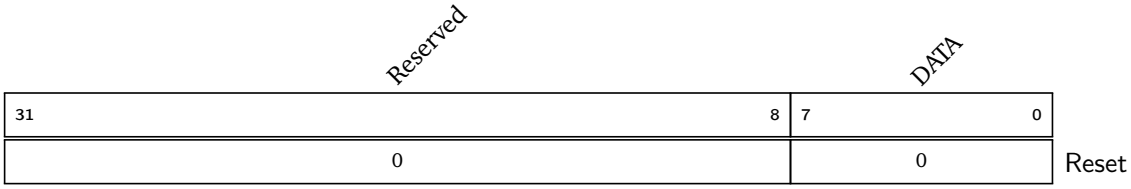
Register C.48: CRC POLYNOMAL REGISTER CRC_POLY *(0xc08)*

POLY This register holds the CRC polynomal up to 32 bits. If the CRC calculation is over less then 32 bits then the redundant bits in the polynomal are not used, e.g. when using 8-bit CRC values, only the lower 8 bit of the polynomal is used.



Register C.49: CRC SEQUENCE REGISTER CRC_SREG *(0xc0c)*

SREG This register holds the calculated CRC value. This register may be written with an initial value (e.g. all ones). Noe that all bits in excess of the CRC calculated value should be ignored, e.g. when calculating 8-bit CRCs the upper bits of SREG should be ignored.



Register C.50: CRC DATA REGISTER CRC_DATA *(0xc10)*

DATA Writing data to this register starts the CRC calculation of this data. Reading returns all zero bits.